Bacterial Foraging Optimization with Leader Selection Strategy for Bi-Objective Optimization

Hong Wang, Yixin Wang, and Yikun Ou *

College of Management, Shenzhen University, Shenzhen 518060, China
Email: ouyikun2019@email.szu.edu.cn

Abstract. Multi-objective problem (MOP) has long been a challenging issue. Many novel Swarm Intelligence (SI) method like Bacterial Foraging Optimization (BFO) has been extended to tackle MOPs recent year. To further improve the efficiency of BFO in multi-objective optimization, this paper proposes a novel BFO for Bi-objective optimization (abbreviated as BIBFO) with enhanced leader selection strategy. The leader selection strategy incorporating with the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) method in comprehensive learning mechanism can direct evolution and enhances the search efficiency. Besides, the strategies of reproduction and elimination are improved using elitism strategy to enhance the collaboration between search group and the external archive, which can speed up the convergence and improve the search efficiency. In addition, the external archive control strategy is further applied to balance the convergence and the solution diversity. The effectiveness of BIBFO is demonstrated on six frequently used benchmarks, and comparative studies have been conducted among bacterial-based multi-objective optimization algorithms. Experimental results indicate that the proposed BIBFO performs well in generation distance (GD) and diversity (Δ) metrics of obtained Pareto front.

Keywords: Bacterial Foraging Optimization, Bi-Objective Optimization, Leader Selection Strategy, DBSCAN

1 Introduction

Many applications in real life often have a multi-objective property. For example, in the stock investment problem, we simply have two conflicting goals. One is spending minimization and the other is revenue maximization or risk minimization [1]. Generally, MOPs contain multiple contradictory objectives, and there is no unique global optimal solution. Therefore, MOPs are usually solved by finding a trade-off solution set. At the same time, MOPs also face more problems, such as computing complexity, dimensionality curse, discontinuous solution distribution, and so on.

Beginning in the 21st century, nature-inspired heuristic methods like evolutionary computing (EC) and swarm intelligence (SI) began to be widely used in MOPs [2]. Some well-known state-of-the-art Multi-objective Evolutionary Algorithms (MOEAs) include the Improving Strength Pareto Evolutionary Algorithm (SPEA2 [3]) and the Non-dominated Sorting Genetic Algorithm II or III (NSGAII [4]， NSGAIII [5]), Multi-objective Artificial Bee Colony Algorithm (MOABC[6]), Multi-objective Particle Swarm Optimization (MOPSO [7]), and so on. The population-based method has excellent global search capabilities [8], and naturally has the advantage of dealing with MOPs. MOEAs are usually able to cover the whole Pareto front in many benchmarks and thus are widely studied. In engineering applications, some MOEAs are proved to have the advantages of less computational burden [9], lower time-consumption [10], stronger robustness [11] and so on. Therefore, it is of great theoretical significance to develop more multi-objective heuristic algorithms.

Bacterial Foraging Optimization (BFO) is a type of swarm intelligence optimization algorithm simulating the foraging behavior of bacteria [12]. It is easy to describe bacterial foraging optimization framework for its simple bionic structure. As BFO has an inner potential to solve the multi-objective optimization algorithm, we design an enhanced BFO algorithm, named Bi-objective Bacterial Foraging Optimization (BIBFO), to fully excavate its optimal performance. In 2013, Wang et al. applied the BFO algorithm to a MOP for the first time (MBFO [13]). Since then, some BFO variants were proposed to solving the MOPs, like a novel MBFO based on a Multi-swarm Cooperative operation (MCMBFO [14]), Multi-objective Bacterial Colony Optimization (MOBCO [11]), Multi-objective BCO with Ring-topology (MORBCO [11]). From their experimental results, these bacterial-based multi-objective optimization algorithms are no less powerful than other swarm intelligence in multi-objective optimization.

Even so, the potential of BFO in MOPs has not been completely realized and it faces several problems like relatively low accuracy, complicated computation. Based on these, the BIBFO has been designed to have a comprehensive learning strategy and an improving process of the reproduction, elimination and dispersal. In addition, to improve the search diversity of the BFO in MOPs and to make the solution closer to the real Pareto front (PF),we presents a leader selection strategy with the basis of Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [15].

The rest of this paper is organized as follows: Section 2 briefly describes the background of MOPs and conventional BFO. The extended BIBFO is introduced in Section 3. In Section 4, the experiments on well-known benchmarks are conducted, followed by an analysis of results obtained. Section 5 presents the conclusions and future works.

2 Related Background

2.1 Multi-Objective Optimization

Multi-objective Optimization Problem (MOP) is abstracted from real life. Mathematically, a MOP can be defined as：

$$\text{Maximize/Minimize } F(x) = \left(f_1(x), f_2(x), f_3(x), \dots, f_m(x)\right)$$
$$Subject\ to{:}\quad g_i(x) \le 0, i = 1,2,3, \dots, q \tag{1}$$

where $x \in \Re^n$ is a n dimensions decision variable including m objectives constrained by q constraint conditions $g(x)$. In fact, it is difficult to obtain the optimal effect on each objective. A solution may be optimal in one objective, but may not be superior in other objectives, which determines that the MOP is pursuing a compromise solution set instead of a certain optimal solution.

2.2 Pareto Optimality

As mentioned above, the goal of the MOP is to find out a relatively satisfactory solution set. Pareto optimality is just such a solution set that represents a balancing result among different objectives. The multi-objective optimization algorithm just tries to approach and cover the true Pareto optimality set. Before using algorithms to solve MOPs, we must understand several concepts below.

Dominance: Suppose the optimization problem is a minimization problem. A vector $\vec{u} = (u_1, u_2, u_3, \dots, u_n)$ is said to dominate $\vec{v} = (v_1, v_2, v_3, \dots, v_n)$ if and only if $u_i \le v_i$, $\forall\, i \in \{1, 2, 3, \dots, n\} \wedge u_j < v_j$, $\exists\, j \in \{1, 2, 3, \dots, n\}$, denoted by $\vec{u} \preccurlyeq \vec{v}$. When $\vec{u}$ is not dominated by any other vectors, so $\vec{u}$ is called a non-dominance solution, also known as a Pareto optimality.

Pareto optimality: For a MOP, a feasible solution $x \in \Re^n$ is called Pareto optimality if and only if $\nexists\, \tilde{x} \in \Re^n$ such that $F(\tilde{x}) \preccurlyeq F(x)$.

Pareto Front: For a MOP, a solution set contained the whole Pareto optimality solution is called a Pareto front (PF), which is usually an equilibrium surface.

2.3 Bacterial Foraging Optimization

Passino[12] firstly proposed BFO in 2002, which is a novel Swarm Intelligence (SI) optimization algorithm simulating bacteria colony foraging behavior. Compared with the behavior of a single bacterium, the bacteria colony can produce swarming effect that helps the whole community gather effectively in areas with high nutrition. The behavior of bacteria foraging mainly includes three processes—Chemotaxis, Reproduction, Elimination and dispersal. The above three processes will be described briefly as follows.

(1) ***Chemotaxis***: Tumbling and swimming are two basic actions of chemotaxis. Supposed that bacteria colony current position is $\theta_i(j, k, l)$, where represents the *i*th bacterium at the *j*th chemotaxis, the *k*th reproduction and the *l*th elimination and dispersal. By tumbling, the bacterium will choose randomly a direction $\Delta(i)$ to explore the area with higher nutrient concentration, where $\Delta(i) \in \Re^n$, $\Delta(i) \in [-1,1]$, indicates the random direction of the *i*th bacterium. By swimming, bacteria get more exploitation power in some direction with more nutrition. In the chemotaxis process, the moving of each bacterium is updated as:

$$\theta_i(j+1, k, l) = \theta_i(j, k, l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \tag{2}$$

where $C(i)$ represents the chemotaxis step size of the *i*th bacterium.

(2) ***Reproduction***: Survival of the fittest is the most basic criterion in the natural world. Only the individual with higher compactivity can survive. Bacteria colony is no exception, which improves the adaptability of the whole community. With

reproduction, the bacteria with higher health fitness can be reproduced and retained in the next iteration, while those one with lower will be removed. The fitness is defined as $J(i,j,k,l)$, which indicates the health status of the *i*th bacterium at the *j*th chemotaxis, the *k*th reproduction and the *l*th elimination and dispersal. In conventional bacterial foraging optimization algorithm, the overall health status of *i*th bacterium can be formulated as：

$$J_i\, health = \sum_{j=1}^{Nc} J(i,j,k,l) \tag{3}$$

(3) ***Elimination and Dispersal***: Due to the natural world is a dynamic and complex environment, some bacteria may die out accidentally or be migrated to another place with a high chance. BFO defines a probability *Ped* that describes a bacterium will be migrated to a newly stochastic place with a certain probability. To some extent, the elimination and dispersal process improves the diversity of the whole bacteria colony and helps escape from the local optimum.

$$\theta_i(j,k,l) = Varmax + (Varmax - Varmin) * rand \tag{4}$$

where *Varmax* and *Varmin* are the boundaries of the decision variables, and the $rand$ is a stochastic value from [0,1].

3 Bi-Objective Bacterial Foraging Optimization

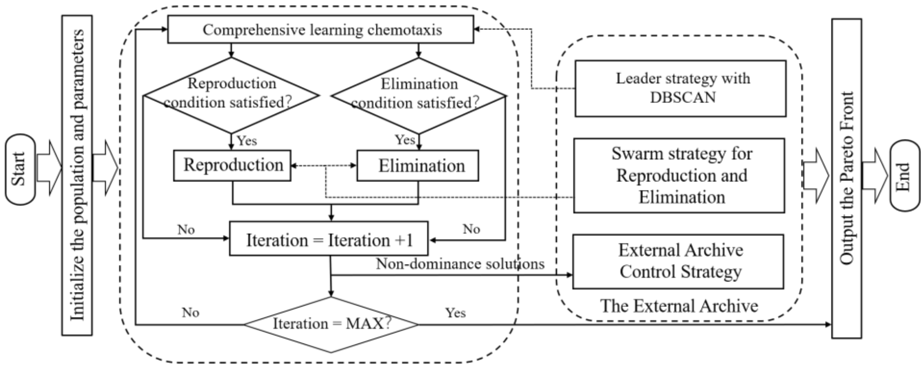

Fig. 1. The overall framework of BIBFO

In this section, the conventional BFO is extended to a bi-objective optimization based on Pareto dominance and the external archive methods. In the proposed BIBFO, novel leader selection, swarm strategy for reproduction and elimination are introduced to fit the bi-objective optimization problem. The DBSCAN is adopted to select a global leader from external archive for chemotaxis in comprehensive learning, which is expected to direct evolution. Unlike the original BFO, the reproduction and elimination-dispersal are performed based on the external archive instead of the evolutionary group. Both reproduction and elimination are based on elitism expected to speed up convergence rate. The overall framework is shown in ***Fig. 1***

3.1 Leader Selection Strategy

The Pareto front of a bi-objective problem usually a group of discrete points in a plane. As the number of non-dominance solutions increases, the distribution and density of the Pareto front also change. To enhance the uniformity of Pareto front, it is necessary to explore more latent solutions in some empty space of current Pareto front. DBSCAN is a typical clustering method based on density distance, by which it can identify the data set distribution and determine different class label [15]. Meanwhile, it does not need to know the clustering number ahead of time and have good clustering ability for irregular sample distribution. Therefore, to improve the capability to search empty solution space, the DBSCAN is applied in the selection of a global optimal solution from the external archive in current iteration for comprehensive learning strategy.

A comprehensive learning mechanism was proposed by Liang [16], and applied in the particle swarm optimization, which has shown great ability in global search and convergence speed. Wang et al. applied a comprehensive strategy in bacterial colony optimization, which blend an operator learning from bacteria colony optimal solution in current iteration[17]. The comprehensive learning mechanism is calculated as follows:

$$\theta_i^t = \theta_i^{t-1} + C(i) * [f_1 * (leader^t - \theta_i^t) + f_2 * \left(\theta_{best\,i} - \theta_i^{t-1}\right)] \tag{5}$$

where θ_i^t is the position of ith bacterium in tth iteration, the f_1 and f_2 are two weight parameters, and $\theta_{\mathrm{best}\,i}$ denotes the history optimum of ith bacterium. $leader^t$ is considered as a leader in the direction of evolution.

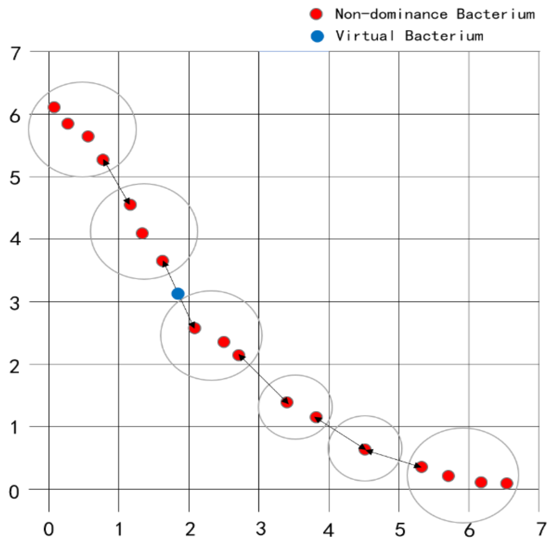

Fig. 2. The leader selection strategy

For MOPs, it is important to note that the goal is to find out as many uniformly distributed Pareto optimality as possible. As shown in ***Fig. 2***, to search for a more even-distributed non-dominance solution set, current non-dominance solutions are clustered by the DBSCAN to several classes. In two adjacent categories with maximum distance, a bacterium that closest to the other category is selected to calculate the leader of current iteration. A virtual bacterium is introduced as a leader to direct evolution, it can be calculated as follows

$$leader^t = \frac{1}{2} * (Position_1^{\mathrm{t}} + Position_2^{\mathrm{t}}) \tag{6}$$

where $Position_{1,2}^t$ is the bacterium selected from the 1st or the 2nd category in *t*th chemotaxis iteration.

3.2 Swarm Strategy

For fully improving the search ability, an adaptive chemotaxis step size is applied in the proposed BIBFO. The core idea is that the initialized bacterium needs a stronger exploration ability in the early stage of searching and more exploitation competence in the late search process[18]. Therefore, the bigger step size is given to initialized bacterium, the more significant the linear decline is with the increase of chemotaxis iteration. The formula is as follows

$$C(i) = C_{min} + \left(\frac{\text{iteration}_{\,max} - \text{iteration}_t}{\text{iteration}_{\,max}}\right) * (C_{max} - C_{min}) \tag{7}$$

where C_{min} and C_{max} represent the minimal and the maximal chemotaxis step size predefined at the start of chemotaxis respectively, $\text{iteration}_{\,max}$ is the maximal number of iterations, iteration_t represents the *t*th iteration.

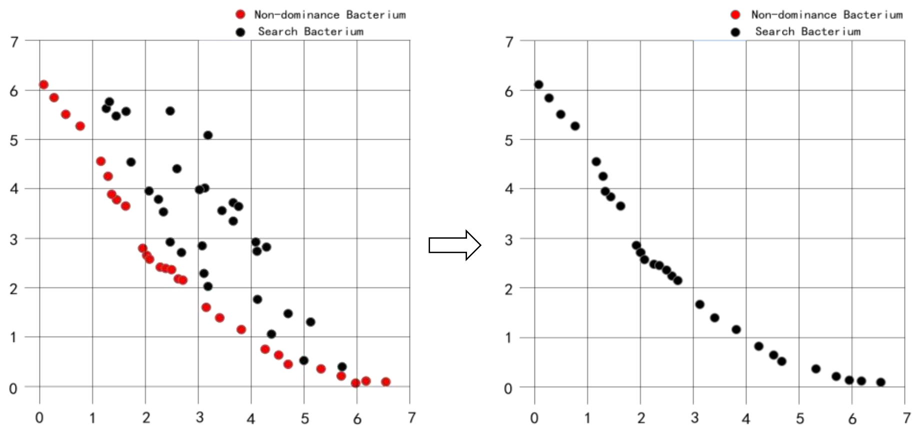

Fig. 3. The reproduction strategy

To retain more excellent bacteria, the reproduction and elimination process were modification based on elitism idea. According to dominance rule, the bacteria of search group will be replaced by the non-dominated ones in external archive when meet the conditions for reproduction, that is, some non-dominance solutions will perform search function after reproduction (see ***Fig. 3***).

Fig. 4 shows the modified elimination process that the bacteria will be replaced by the neighborhood of a randomly-chosen non-dominated solution in current iteration with a predefined probability.

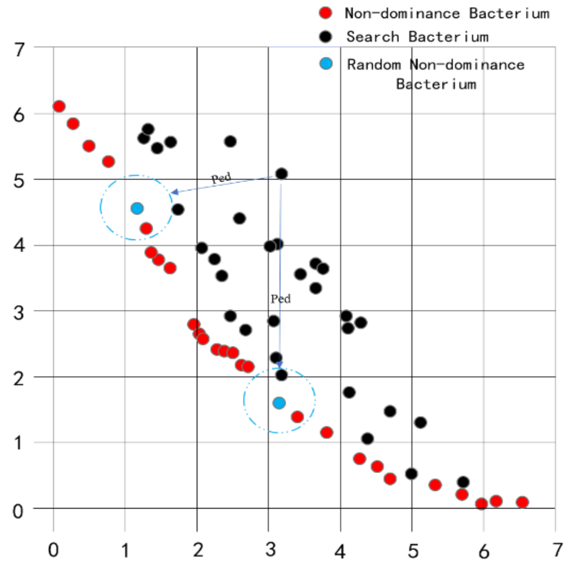

Fig. 4. The elimination strategy

3.3 External Archive Control

Colello[7] firstly applied an adaptive external archive to store the non-dominance solution in multi-objective particle swarm optimization. As the search progresses, an increasing number of non-dominance solution is found out while the external archive size is limited. Thus, it is necessary to apply some strategies to control the external archive to obtain high diversity. In the proposed BIBFO, the crowding distance method[4] is applied to identify the density of the non-dominance solution distribution. The strategy is that the bacteria that are in the low-density area have a better chance of being preserved than bacteria that are in a higher one. After calculating the crowding distance of each bacterium, the bacteria were sort in an order according to their crowding distance and the bacterium with the highest crowding distance will be deleted.

Aim to achieve the more uniformly-distributed Preto Front, the proposed BIBFO will not allow the dominated solutions with the same cost enter into the external archive. To be specific, there are serval bacteria may consum the same cost value due to the SI property, the search members will learn from the best one. For a Pareto method, the solutions with the same cost value are redundant and may decrease the diversity of final Pareto set. In order to avoid these adverse effects, only one in the solutions with the same cost value will be selected to preserve in external archive. The pseudo code of BIBFO is shown in the ***Table 1***.

Table 1. The pseudo-code of BIBFO

```
Algorithm: BIBFO
Initialize: Bacteria's position, the parameters value, etc.
For its = 1: the maximum number of chemotaxis MaxIts
    For each bacterium
        Updating position with chemotaxis operator using (5)
        Updating the bacterium history best using the non-dominance rule
    End For
    Updating the external archive, do
        Adding non-dominance bacteria into the external archive
        Controlling the external archive with crowding distance method
    Updating the global optimum leader with DBSCAN clustering algorithm
    When satisfies the reproduction condition, do
        Reproduce operation
    When satisfies elimination and dispersal condition, do
        Eliminate operation
    its ← its + 1
End For
Output：Pareto front
```

4 Experiments and Results

4.1 Performance Metrics

To evaluate the performance of a multi-objective optimization algorithm, the generation distance (GD) and the diversity (Δ) are introduced in this paper.

(1) ***Generation distance (GD)***: *GD* is a distance-based measurement, which estimates the distance between the actual Pareto front and the current Pareto optimality set[19]. The smaller the value obtains, the better performance is. It is calculated as

$$GD = \frac{\sqrt{\sum_{i=1}^{n} distance_i^2}}{n} \tag{8}$$

where n is the number of the current Pareto optimality set and $distance_i$ denotes the Euclidean distance between the ith solution of current Pareto optimality set and the nearest Pareto optimality of the actual Pareto front.

(2) ***Diversity (Δ)***: Diversity (Δ), is a metric reflecting the extent of the obtained solution set[4]. The smaller the value Δ gets, the better the diversity is, the better the performance of the optimization algorithm. It is formulated as

$$\Delta = \frac{d_f + d_l + \sum_{s=1}^{n-1} \left|d_s - \bar{d}\right|}{d_f + d_l + (n-1)\bar{d}} \tag{9}$$

where d_s is the Euclidean distance between sequential solutions in the current Pareto optimality set and $\bar{d}$ is the average of them. Besides, d_f and d_l are the Euclidean distances between the lower and upper boundary solutions of the current Pareto optimality set and the extreme solutions of the actual Pareto front.

4.2 Problems and Algorithm Settings

Six well-known benchmarks including Zitzler studies (ZDT1~3[20]), Schaffer (SCH1[21]) , Fonseca (FON[22]) and Kursawe (KUR[23]), are chosen to test the proposed BIBFO, and all of them are bi-objective problems with no constraints. To verify that BIBFO improves the performance of traditional BFO extended to MOPs, 3 multi-objective optimization algorithms based on bacteria colony are selected for comparing experiments, including MORBCO[11], MCMBFO[14], MOBCO[11]. All of these comparing algorithms, the population size and the external archive size are 100, the maximum number of iterations is set as 1000.

As for the proposed BIBFO, the parameters are as follows: *npop* = 100, C_{min}=0.05, C_{max} = 1.2, *MaxIts* =300, f_1 = 3, f_2 = 1, *Ped* = 0.5, *Ns* = 5. As the DBSCAN, the ε representing the neighbor threshold and *Minpts* describing the minimum number of samples in a class, are set 0.02 and 1 respectively. All test conduct 30 times independently.

4.3 Results and Analysis

The comparison results are shown as ***Table 2***. It shows the performance scores of the test algorithms on the generation distance and diversity of each benchmark. The best results obtained by algorithms have been highlighted in bold. It can be concluded that the average performance of BIBFO do better than the other compared algorithms in terms of these benchmarks.

Table 2. Comparison of performance metrics on benchmarks

		Generation distance (GD)				***Diversity (Δ)***			
		Best	Worst	Mean	Std.	Best	Worst	Mean	Std.
ZDT1	BIBFO	**3.47E-04**	**7.83E-04**	**4.89E-04**	**1.13E-04**	**1.56E-01**	**2.76E-01**	**2.08E-01**	**2.92E-02**
	MORBCO	1.37E-03	7.00E-03	3.60E-03	1.30E-03	4.75E-01	8.30E-01	6.26E-01	7.77E-02
	MCMBFO	7.90E-03	9.00E-03	8.40E-03	3.86E-04	6.09E-01	6.96E-01	6.54E-01	3.33E-02
	MOBCO	1.22E-01	5.55E-01	2.72E-01	9.91E-02	4.18E-01	9.24E-01	6.88E-01	1.18E-01
ZDT2	BIBFO	**4.00E-04**	**8.51E-04**	**5.39E-04**	**1.10E-04**	**1.52E-01**	**3.17E-01**	**2.63E-01**	3.70E-02
	MORBCO	1.05E-03	9.74E-03	3.86E-03	2.30E-03	5.94E-01	8.48E-01	6.80E-01	6.90E-02
	MCMBFO	8.80E-03	1.40E-02	1.18E-02	1.90E-03	6.06E-01	6.41E-01	6.21E-01	**1.37E-02**
	MOBCO	1.41E-02	5.68E-02	3.11E-02	1.03E-02	5.51E-01	8.39E-01	6.84E-01	7.03E-02
ZDT3	BIBFO	**7.15E-04**	**1.17E-03**	**9.40E-04**	**1.20E-04**	**3.33E-01**	**6.53E-01**	**4.72E-01**	8.48E-02
	MORBCO	3.45E-03	1.58E-02	6.70E-03	3.01E-03	5.60E-01	9.60E-01	7.32E-01	8.85E-02
	MCMBFO	6.30E-03	7.10E-03	6.70E-03	2.99E-04	5.58E-01	6.62E-01	6.23E-01	4.40E-02
	MOBCO	7.63E-02	2.53E-01	1.50E-01	4.77E-02	4.85E-01	8.36E-01	6.49E-01	8.74E-02
SCH1	BIBFO	**7.58E-04**	**1.08E-03**	**9.34E-04**	**7.21E-05**	**1.59E-01**	**2.29E-01**	**1.92E-01**	**1.68E-02**
	MORBCO	5.07E-03	8.46E-03	6.61E-03	9.11E-04	5.02E-01	6.32E-01	5.66E-01	3.56E-02
	MCMBFO	5.80E-03	7.20E-03	6.40E-03	5.75E-04	4.97E-01	6.40E-01	5.37E-01	4.16E-02
	MOBCO	5.13E-03	1.04E-02	7.00E-03	1.14E-03	5.03E-01	6.42E-01	5.64E-01	3.67E-02
FON	BIBFO	**2.92E-04**	**4.94E-04**	**3.89E-04**	**6.65E-05**	**1.60E-01**	**3.13E-01**	**2.22E-01**	3.98E-02
	MORBCO	1.31E-03	2.54E-03	1.98E-03	2.98E-04	5.40E-01	6.84E-01	5.98E-01	**3.66E-02**
	MCMBFO	3.30E-03	3.90E-03	3.60E-03	5.80E-04	5.32E-01	5.87E-01	5.62E-01	3.78E-02
	MOBCO	3.00E-03	4.17E-03	3.53E-03	2.98E-04	4.97E-01	6.86E-01	5.85E-01	4.31E-02
KUR	BIBFO	**7.36E-03**	**2.69E-02**	**1.23E-02**	4.47E-03	7.02E-01	9.59E-01	8.48E-01	6.13E-02
	MORBCO	1.74E-02	4.91E-02	2.73E-02	7.45E-03	**5.14E-01**	7.72E-01	6.48E-01	6.51E-02
	MCMBFO	2.38E-02	3.24E-02	2.89E-02	**3.70E-03**	6.67E-01	8.56E-01	7.61E-01	6.87E-02
	MOBCO	2.95E-02	6.39E-02	4.23E-02	8.43E-03	5.20E-01	**6.66E-01**	**5.99E-01**	**3.35E-02**

As we have seen from ***Table 2***, whether it is diversity or generation distance, the performance metrics of BIBFO are much better than other algorithms, which can well reflect that the distance between the true Pareto front and the Pareto optimal set obtained by BIBFO is closer. Besides, the stability of BIBFO is also far superior to comparison algorithms. To our knowledge, these results indicate that the proposed BIBFO further improve the capability of BFO to tackle MOPs.

It is noted that the MOBCO is the weakest performance in any problems. On the contrary, the MORBCO, which introduced a ring topology for bacterial communication based on MOBCO, improved the performance to a certain extent. The MOBCO almost defeated the MCMBFO incorporated multi-swarm cooperative operation among six benchmarks. It can be concluded that effective communication strategy and collaboration strategy can greatly improve the performance of raw algorithm.

Due to the space limitation, only the optimal Pareto fronts gained by BIBFO were displayed as Fig. 5. It can be observed that most the optimal Pareto fronts are close to the true Pareto front. For specific problem, the decision dimensions of SCH1, FON, and KUR are only 1 or 3, and the search range is relatively small. SCH1 and FON almost cover the true Pareto front fully in each comparing algorithm. However, note that the proposed algorithm performed poorly on the KUR function. The reason is that the true Pareto front of the KUR function is discontinuous, which makes BIBFO choose a wrong leader using the DBSCAN clustering method.

ZDT1~3 are relatively complicated, there are 30 dimensions in decision variable, the search space is relatively large. From the experimental results, the BIBFO can also achieve the expected results on ZDT1~3 and outperforms other algorithms. However, the performance of ZDT3 is slightly poor comparing to ZDT1 and ZDT2 for the partitioned Pareto front distribution.

5 Conclusions and Feature Work

In this paper, we proposed a bacterial foraging optimization algorithm with multi-strategy for bi-objective optimization. To be specific, leader selection operation using DBSCAN incorporated into comprehensive learning chemotaxis, helps the BIBFO clarify the evolution direction and thus speed up the convergence. Besides, the swarm strategy, including linear decreasing chemotaxis step size, modified reproduction and elimination based on elitism, plays the key role on enhancing the search capability. Comparing the other bacterial-based multi-objective algorithm, the modified external archive strategy of BIBFO further excavated the role of the external archive and improved the storage efficiency of non-dominated solutions.

Then the compared experiments were conducted and proved that the proposed BIBFO algorithm performs well on diversity and generation distance metrics of several bi-objective benchmarks, which achieved the expected improvement effect. Results proved that using BFO with multi-strategy are effective in enhancing the performance of solving bi-objective problems.

Even so, BIBFO itself has its limitations. Compared with other swarm intelligence algorithms, a big computation task cannot be ignored and it has multiple controllable parameters that need to be adjusted according to different problems. In future work, we

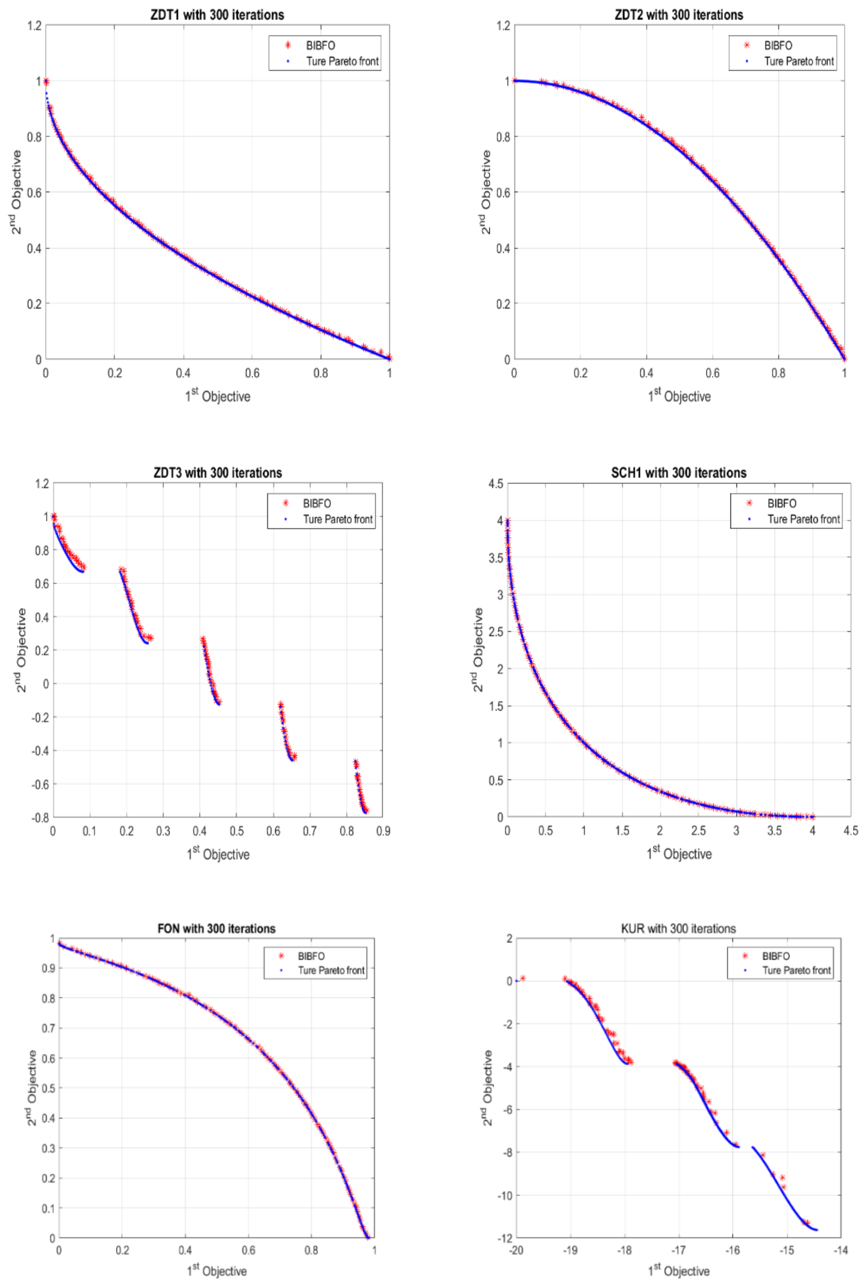

Fig. 5. The Pareto front obtained by BIBFO

will continue to improve the structure of the bacterial-based multi-objective algorithm and enhance the effectiveness of multi-strategy to solve more complex MOPs and applications.

Acknowledgment. This work is partially supported by The National Natural Science Foundation of China (Grants Nos. 71901152, 71971143), Natural Science Foundation of Guangdong Province (2020A1515010752), and Natural Science Foundation of Shenzhen University (85303/00000155).

References

1. Babaei, G., Bamdad, S.: A multi-objective instance-based decision support system for investment recommendation in peer-to-peer lending. Expert Syst Appl. 150, 113278 (2020).
2. Zhou, A., Qu, B.Y., Li, H., Zhao, S.Z., Suganthan, P.N., Zhangd, Q.: Multiobjective evolutionary algorithms: A survey of the state of the art. Swarm Evol Comput. 1, 32–49 (2011).
3. Eckart Zitzler, Marco Laumanns, and L.T.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics). 8710 LNCS, 35–46 (2001).
4. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput. 6, 182–197 (2002).
5. Yuan, Y., Xu, H., Wang, B.: An improved NSGA-III procedure for evolutionary many-objective optimization. GECCO 2014 - Proc 2014 Genet Evol Comput Conf. 661–668 (2014).
6. Zou, W., Zhu, Y., Chen, H., Zhang, B.: Solving multiobjective optimization problems using artificial bee colony algorithm. Discret Dyn Nat Soc. 2011, (2011).
7. Coello, C.A.C.: Handling Multiple Objectives With Particle Swarm Optimization. IEEE Trans Evol Comput. 15, 45–82 (2014).
8. Fister, I., Yang, X.S., Brest, J., Fister, D.: A brief review of nature-inspired algorithms for optimization. Elektroteh Vestnik/Electrotechnical Rev. 80, 116–122 (2013).
9. Han, H., Lu, W., Qiao, J.: An Adaptive Multiobjective Particle Swarm Optimization Based on Multiple Adaptive Methods, (2017).
10. Li, X., Wong, K.C.: Multiobjective Patient Stratification Using Evolutionary Multiobjective Optimization. IEEE J Biomed Heal Informatics. 22, 1619–1629 (2018).
11. Niu, B., Liu, Q., Wang, Z., Tan, L., Li, L.: Multi-objective bacterial colony optimization algorithm for integrated container terminal scheduling problem. Nat Comput. 3, (2020).
12. Passino, K.M.: Biomimicry of bacterial foraging for distributed optimization and control. IEEE Control Syst Mag. 52–67 (2002).
13. Niu, B., Wang, H., Wang, J., Tan, L.: Multi-objective bacterial foraging optimization. Neurocomputing. 116, 336–345 (2013).
14. Niu, B., Liu, J., Tan, L.: Multi-swarm cooperative multi-objective bacterial foraging optimisation. Int J Bio-Inspired Comput. 13, 21–31 (2019).

15. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Proc 2nd Int Conf Knowl Discov Data Min. 226–231 (1996).
16. Huang, V.L., Suganthan, P.N., Liang, J.J.: Comprehensive learning particle swarm optimizer for solving multiobjective optimization problems. Int J Intell Syst. 21, 209–226 (2006).
17. Niu, B., Wang, H.: Bacterial colony optimization. Discret Dyn Nat Soc. 2012, (2012).
18. Niu, B., Fan, Y., Zhao, P., Xue, B., Li, L., Chai, Y.: A novel bacterial foraging optimizer with linear decreasing chemotaxis step. Proc - 2010 2nd Int Work Intell Syst Appl ISA 2010. 1–4 (2010)..
19. Van Veldhuizen, D.A., Lamont, G.B.: Multiobjective evolutionary algorithms: analyzing the state-of-the-art. Evol Comput. 8, 125–147 (2000).
20. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: empirical results. Evol Comput. 8, 173–195 (2000).
21. Schaffer, J.D.: Multiple objective optimization with vector evaluated genetic algorithms. 1st Int Conf Genet Algorithms. 93–100 (1985).
22. Fonseca, C.M., Fleming, P.J.: Multiobjective optimization and multiple constraint handling with evolutionary algorithms - Part I: A unified formulation. IEEE Trans Syst Man, Cybern Part ASystems Humans. 28, 26–37 (1998).
23. Kursawe, F.: A variant of evolution strategies for vector optimization. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics). 496 LNCS, 193–197 (1991).